

Лабораторная работа № 2

Вариант Е-1

При выполнении лабораторной работы не допускается использовать сторонние библиотеки, за исключением `re`.

Задание 1

На вход программы подается 2 строки.

Первая строка содержит информацию о людях, представленную в формате: `first_name=<имя>,second_name=<отчество>,last_name=<фамилия>,phone=<телефон>,email=<email>;first_name2=<имя>,second_name2=<отчество>,last_name2=<фамилия>,phone2=<телефон>,email2=<email>;...`

Вся информация на английском языке. Телефон состоит из 10 цифр, идущих подряд (все, кроме кода страны).

Необходимо учесть, что данная информация будет передаваться в формате номеров символов в таблице ASCII, разделенных символом "#", например вместо "name" будет введено "110#97#109#101". В таком формате закодирована вся строка, содержащая данные о людях, в том числе разделительные символы. Данные необходимо преобразовать к символьному (читаемому) виду, преобразовать к словарю и сохранить в виде списка словарей данные о людях. В случае, если какие-либо данные не были введены (в том числе не указан один из ключей), они должны иметь значение пустой строки в словаре. Также, если будут указаны ключи, которые не предусмотрены программой – их вместе с соответствующим значением необходимо исключить.

Вторая строка содержит информацию для поиска человека.

Вся часть по вводу и обработке пользовательского ввода должна быть вынесена в отдельную функцию. Можно выделять логику программы в дополнительные функции.

Далее необходимо реализовать функцию, принимающую в качестве параметров список людей (полученных в результате обработки первой строки), а также поисковую строку (вторая строка пользовательского ввода). Данная функция должна фильтровать людей по следующим полям: отдельно фамилия, имя, отчество, номер телефона, адрес электронной почты, ФИО целиком, а также ФИО в формате Фамилия И.О.

Функция должна возвращать список подходящих под поисковую строку людей в качестве результата.

Необходимо продемонстрировать работоспособность программы для случая работы с информацией об одном или нескольких людях.

* Дополнительно рекомендуется реализовать функцию для создания тестовых данных. Функция должна позволять преобразовывать текстовые строки к следующему формату: номера символов в таблице ASCII, разделенных символом "#".

Пример:

Для одного человека

Ввод:

```
102#105#114#115#116#95#110#97#109#101#61#70#101#108#105#112#97#4  
4#108#97#115#116#95#110#97#109#101#61#67#111#108#108#97#116#111#  
110#44#101#109#97#105#108#61#119#99#111#108#108#97#116#111#110#5  
0#64#121#97#104#111#111#46#99#111#109#44#112#104#111#110#101#61#  
51#57#49#55#49#51#51#55#52#55#44#115#101#99#111#110#100#95#110#9  
7#109#101#61#87#105#108#108#105#101
```

После преобразования к читаемому виду:

```
first_name=Felipa, last_name=Collaton, email=wcollaton2@yahoo.com,  
phone=3917133747, second_name=Willie
```

После преобразования к словарю:

```
{  
'first_name': 'Felipa', # Имя  
'last_name': 'Collaton', # Фамилия  
'email': 'wcollaton2@yahoo.com',  
'phone': '3917133747',  
'second_name': 'Willie' # Отчество  
}
```

Поисковые строки, под которые подходит данный пользователь:

```
"lipa", "Collaton", "Collaton Felipa", "collaton Felipa W",  
"2@yahoo", "337", "Collaton F.W", "Felipa"
```

Задание 2

Написать рекурсивную функцию, определяющую, является ли введенное пользователем число n степенью числа k . Числа n , k - неотрицательные целые числа, введенные пользователем. Вернуть True, если является, False - в ином случае.

Пример:

n=32, k=2 ответ: True (*так как $32 == 2^5$*)
n=1, k=10000 ответ: True (*так как $1 == 10000^0$*)
n=25, k=4 ответ: False
n=0, k=0 ответ: True (*так как $0 == 0^x$*)